Malaysian Institute of Microelectronics System (MIMOS Berhad), Malaysia.

AccLib

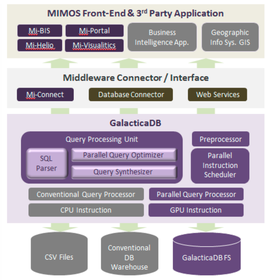
# Massively Parallel Geo-Spatial Coordinates Computation with GalacticaDB

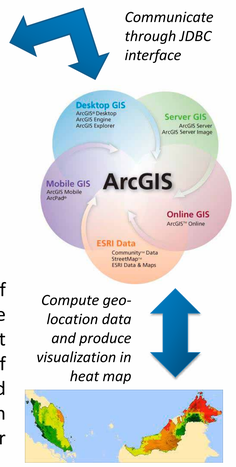
## ABSTRACT

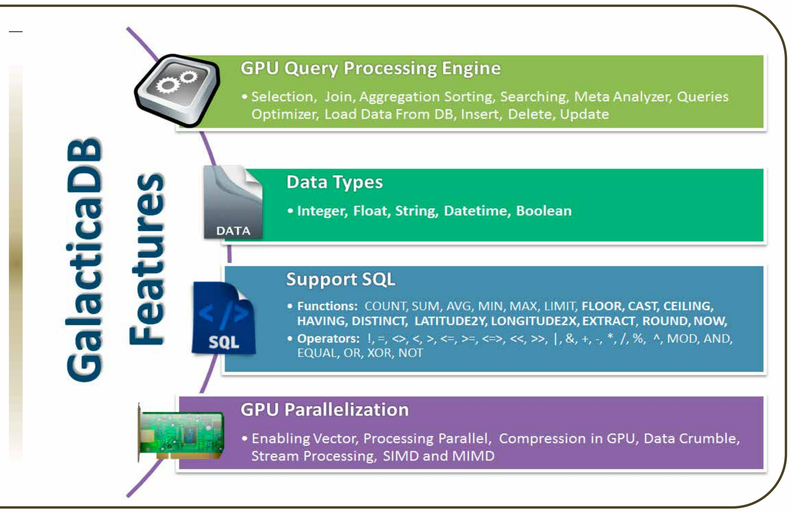
Persons or things has become a mobile sensor, with GPS enabled. These self-quantified technologies generate humongous of raw data, which can turn into valuable information for users and business. There is even a scale that will tweet your weight. Incredibly, every minute there are 100,000 tweets includes location either explicitly or via textual analysis. This has emphasize the need for developing massive computationally efficient geospatial analytics tailored for turning big volume raw data to business value. We proposed a massively parallel SQL-like parallel engine, GalacticaDB with extended geo-spatial capabilities. It is an emerging GPU database engine that accelerating analytical computation with parallelizing queries processing and exploiting NVIDIA Tesla GPUs. Our results indicate that the GPU is an effective and energy efficient co-processor for executing database operations.

## SYSTEM DESIGN

*Database connector* enables GalacticaDB to connect to any database to perform frontend application interaction, data extraction, query operation and data interchange. *Query processing unit* carries out processes of query investigation by performing the basic processing and positioning operations. It targets to break down the input user queries to further clause of objects and collects the decomposed SQL objects to determine either CPU or GPU executable instructions to be used. Then, it produces an execution query plan. There is further adjustment of the plan by analyzing and tracing parallelizable points and rearranges clause of objects execution. GalacticaDB execution engine performs the accelerated query execution on either CPU or GPU

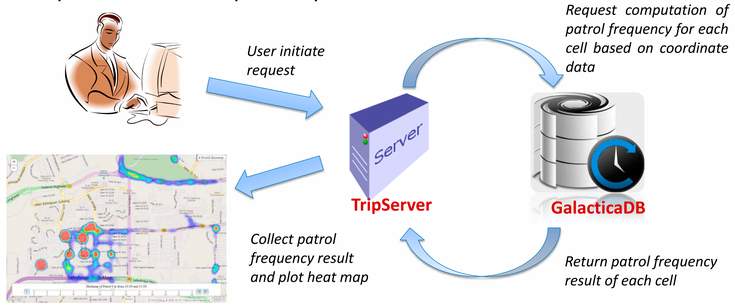



## USE CASE

Calculating frequency of the streets, which had been patrolled by police cars. Heat map is generated with 2 hours time interval location data of everyday of the month. Computation performance is compared between Apache Spark and GalacticaDB.




## ENVIRONMENT

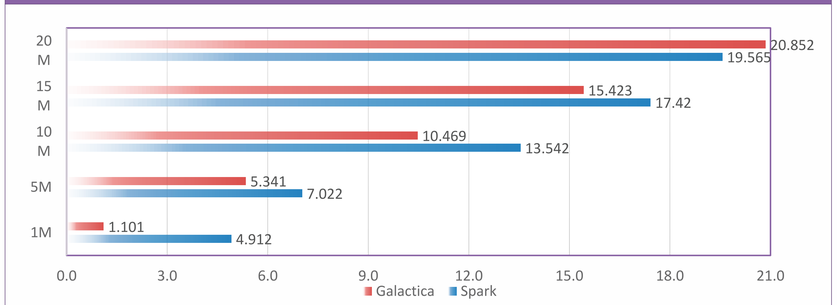

**Test Data**

- Using police car patrol database from April of 2010 to 2013
- Performing same queries on various size of dataset
- Comparing Galactica against Apache Spark

**Environment**

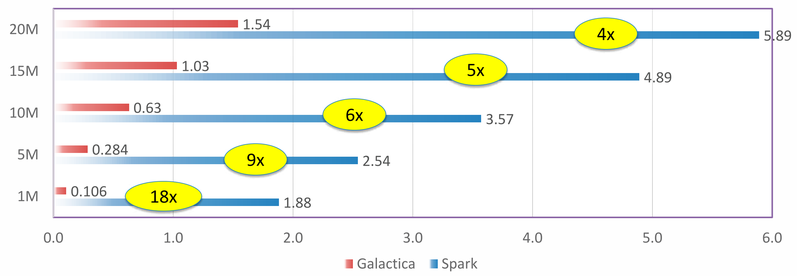
Intel Xeon X5680 (6 cores) x 2 (Total 12 cores)
NVIDIA Tesla K20c
22 GB RAM
WD HDD 1 TB
Windows 7 Professional 64-bits

## RESULT




*Graph 1: Preprocessing Data*

Input CSV file is processed into corresponding columnar data format (Spark – Parquet file, GalacticaDB – GPU File System). As observed, Spark has a minimum startup overhead to process CSV file. Spark eventually overtakes GalacticaDB due to the use of multiple threads to process CSV file.




*Graph 2: Heat map computation*

Computation time which contains huge amount of numerical computations to plot police car patrol heat map is being compared between Spark and GalacticaDB. GalacticaDB has minimum computation startup overhead compare to Spark. As number of records increases, time taken for computation for GalacticaDB grow much slower than Spark. Graph 2 shows that GalacticaDB performs over 4x faster than Spark on numerical computations. GalacticaDB performance is good enough to provide timely visualization on geo-location data.

## LESSONS LEARNED

- GPU parallel processing performs faster than Spark on multicore CPU in terms of numerical computation
- Result shows that there good linear scaling with larger dataset
- Low cost GPU workstation has a competitive result against Map-Reduce cluster

## FUTURE WORK

- Support SQL operations with geodatabase
- Enable capability to use multiple GPUs on a single workstation
- Exploring distributed queries execution across GPU cluster

## ACKNOWLEDGEMENT

This research was done under Joint Lab of "NVIDIA - HP - MIMOS GPU R&D and Solution Center". This is the first GPU solution center in South East Asia established in October 2012. Funding for the work came from MOSTI, Malaysia. The authors would like to thank Prof. Simon See and Pradeep Gupta from NVIDIA for the supports.

MIMOS Innovation for Life™

**http://gpu.mimos.my**
**http://atl.mimos.my**
**http://www.mimos.my**

**References:**

1. P. K. Chong, Ettikan K. Karuppiah and K.K. Yong, "A Multi-GPU Framework for In-Memory Text Data Analytics", The 27th IEEE International Conference on Advanced Information Networking and Applications (AINA), Barcelona, Spain, Mar. 2013.
2. K. K. Yong, E. K. Karuppiah, and S. See, "Galactica : A GPU Parallelized Database Accelerator," in International Conference on Big Data Science and Computing, ASE 2014, 2014.
3. E. K. Karuppiah, K. K. Yong, and K. Singh, "A Middleware Framework for Programmable Multi-GPU-Based Big Data Applications," in GPU Computing and Applications, Y. Cai and S. See, Eds. Singapore: Springer Singapore, 2015, pp. 187–206.